

FILE MANAGEMENT SYSTEM

Submitted by

Name of the Students: Pratik Chakraborty

Enrolment Number: 12022002003208

Section: H

Class Roll Number: 67

Stream: ECE

Subject: Programming for Problem Solving using C

Subject Code: ESC103(Pr.)

Department: Basic Science and Humanities

Under the supervision of
Swarnendu Ghosh

Academic Year: 2022-26

PROJECT REPORT SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE SECOND SEMESTER



**DEPARTMENT OF BASIC SCIENCE AND HUMANITIES
INSTITUTE OF ENGINEERING AND MANAGEMENT, KOLKATA**



CERTIFICATE OF RECOMMENDATION

We hereby recommend that the project prepared under our supervision by **Pratik Chakraborty**, entitled **FILE MANAGEMENT SYSTEM** be accepted in partial fulfillment of the requirements for the degree of partial fulfillment of the second semester.

Head of the Department
Basic Sciences and Humanities
IEM, Kolkata

Project Supervisor

1 Introduction

1.1 Objective:-

The main objective of this project is to develop a file management system in which can create, modify or delete a file.

1.2 Organization of the Project

The project is organized into 3 parts or 3 cases.

- 1) In the first case the program runs and creates a txt file.
- 2) In the second case we can modify the said file.
- 3) In the third case we can delete the file.
- 4) The user can choose what to do from the above options.

1.3 Variable, Functions, Statements descriptions:-

- 1) int,char etc common variables are used to enter the number and text input
- 2) Other than that If else, switch case etc are used to determine the output of the user
- 3) printf, scanf is used to take user input and print it.

2 Programs

```
#include <stdio.h>
#include <stdlib.h>

int main() {
    int choice;
    char filename[50];
    FILE *fp;

    do {
        printf("Select an option:\n");
        printf("1. Create a file\n");
        printf("2. Delete a file\n");
        printf("3. Modify a file\n");
        printf("4. Exit\n");
```

```
scanf("%d", &choice);

switch(choice) {
    case 1:
        printf("Enter the name of the file to create: ");
        scanf("%s", filename);
        fp = fopen(filename, "w");
        if (fp == NULL) {
            printf("Error creating file\n");
        } else {
            printf("File created successfully\n");
            fclose(fp);
        }
        break;
    case 2:
        printf("Enter the name of the file to delete: ");
        scanf("%s", filename);
        if (remove(filename) == 0) {
            printf("File deleted successfully\n");
        } else {
            printf("Error deleting file\n");
        }
        break;
    case 3:
        printf("Enter the name of the file to modify: ");
        scanf("%s", filename);
        fp = fopen(filename, "a");
        if (fp == NULL) {
            printf("Error modifying file\n");
        } else {
            printf("File modified successfully\n");
            fclose(fp);
        }
        break;
    case 4:
        printf("Exiting program...\n");
        break;
    default:
        printf("Invalid choice\n");
}
```

```
    } while (choice != 4);
```

```
    return 0;
```

```
}
```

3) Outputs

The screenshot shows the Visual Studio Code interface with the title bar "C programming". The left sidebar shows a folder named "C PROGRAMMING" containing various files like .vscode, a.exe, brc.c, bro.exe, C_cc, etc. The main editor area displays a C program with code for file deletion and modification. The terminal tab at the bottom shows the command "PS C:\Users\iem\Desktop\C programming> gcc C_cc.c" followed by a menu selection:

```
PS C:\Users\iem\Desktop\C programming> gcc C_cc.c
PS C:\Users\iem\Desktop\C programming> .\a.exe
Select an option:
1. Create a file
2. Delete a file
3. Modify a file
4. Exit
```

This screenshot is similar to the one above, but the terminal output has changed. It shows the user selecting option 3 (Modify a file), entering "new.txt" as the filename, and then seeing the message "File modified successfully". The terminal then displays the menu again:

```
4. Exit
3
Enter the name of the file to modify: new.txt
File modified successfully
Select an option:
1. Create a file
2. Delete a file
3. Modify a file
4. Exit
4
Exiting program...
PS C:\Users\iem\Desktop\C programming>
```

Screenshot of VS Code interface showing a C programming project. The Explorer sidebar shows files like .vscode, a.exe, b.exe, c.cc, etc. The main editor shows a C program with code for deleting and modifying files. The terminal shows the command to compile and run the program.

```
PS C:\Users\ien\Desktop\C programming> gcc c.c
PS C:\Users\ien\Desktop\C programming> .\a.exe
Select an option:
1. Create a file
2. Delete a file
3. Modify a file
4. Exit
```

Screenshot of VS Code interface showing a C programming project. The Explorer sidebar shows files like .vscode, a.exe, b.exe, c.cc, etc. The main editor shows the same C program as the first screenshot. The terminal shows the creation of a new file named new.txt.

```
2. Delete a file
3. Modify a file
4. Exit
1
Enter the name of the file to create: new.txt
File created successfully
Select an option:
1. Create a file
2. Delete a file
3. Modify a file
4. Exit
```

